



## UvA-DARE (Digital Academic Repository)

### Making sense of legal texts

de Maat, E.

**Publication date**  
2012

[Link to publication](#)

#### **Citation for published version (APA):**

de Maat, E. (2012). *Making sense of legal texts*. [Thesis, fully internal, Universiteit van Amsterdam].

#### **General rights**

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

#### **Disclaimer/Complaints regulations**

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

## 4 Classification of Sentences

Most laws are too large to model in a single step. Therefore, we will first model parts of the law, and then integrate those parts in order to come to a complete model. As a unit for those partial models we chose the sentence, as this is the basic building block for the text.

In order to choose the right approach for modelling the sentence, we first want to know its general meaning: is it a normative sentence, or a sentence that changes an existing law? Depending on this general meaning, it is likely that a different sort of model should be created.

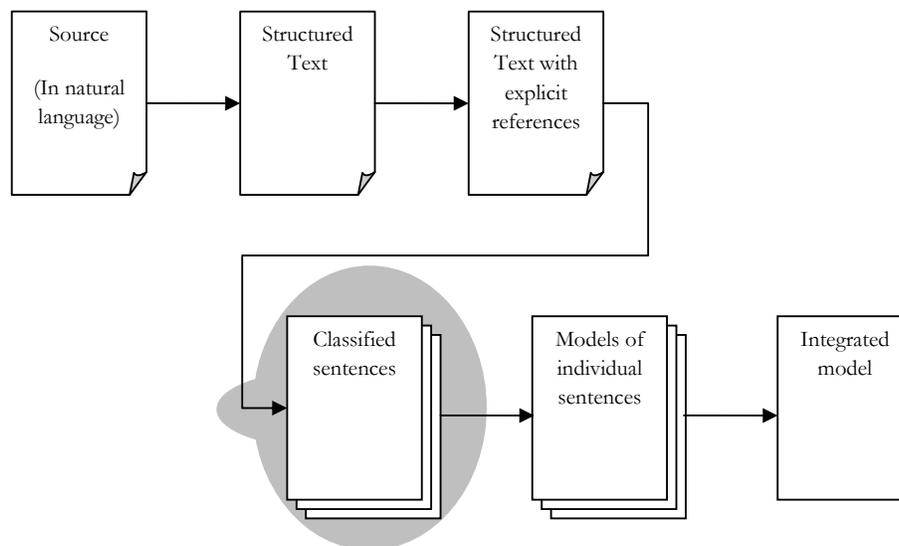


Figure 6: Classifying sentences step

Figure 6 shows the position of this step in the entire process. The input for this step is a document in which the structure has been marked up, so that we can target the individual sentences. After we have classified the sentences in this step, we will continue to create the models of the individual sentences. As we take this step before analysing the sentences in detail, we wish to make the classification using a shallow approach, looking at the surface structure of the sentence only, without doing an elaborate analysis. We have tried two different approaches: a knowledge engineering approach using patterns, and a machine learning approach using bags of words (i.e. all the individual words in a sentence). Deschamps (2011) found that laws in the Dutch language lack uniformity, which makes classification more difficult. Still, our research shows that classification is still feasible (see section 4.5).

However, before we classify the sentences, we need to know what kinds of sentences exist. Legal theory often discusses the kind of rules or norms that may occur in a law, and such classifications have been used for formal models as well (see for example Sartor, 2006). However, these classifications seldom address the actual text. The official guidelines for legislative drafting and legislative drafting literature describe some sentence types (which we will encounter later in this chapter), but do not provide an overall categorisation.

Still, there are two categorisations we wish to present here. First of all, our own broad categorisation (published before in de Maat and Winkels, 2007 and de Maat, Winkels and van

Engers, 2009), and two more detailed categorisations from Tiscornia and Turchi (1997) and Atienza and Manero (1998).

In our own vision, the law consists of a set of core rules and several layers of supporting rules. The goal of regulations is (or perhaps: should be) to set rules for the people living in a country<sup>60</sup> and the organisations established in that country. The regulations tell them what they can do and what they cannot do, and what their rights and obligations are. So, we could expect a regulation to mainly consist of statements like *Everybody has the right to freedom of speech* and *If you take care of a child less than eighteen years of age, you have a right to child subsidy*. Such statements do indeed appear in the law, for example:

**General Child Benefit Law, article 7, sub 1<sup>61</sup>**

Conform the stipulations of this law, the insured has a right to child benefit for an own child, a stepchild and a foster child which:

- a. is younger than 16 years of age and belongs to his household; or
- b. is younger than 18 years of age and is maintained by him for a significant amount.

However, in addition to such core rules, there are additional rules that help support these core rules. By merely specifying that people have a right to child benefit, these benefits are not automatically distributed. A system needs to be set up for this. This leads to two layers of additional procedural overhead. A first layer, directed at the behaviour of citizens (or their organisations), tells citizens what procedure they have to follow to achieve certain goals. An example of this is:

**General Child Benefit Act, article 14, sub 2<sup>62</sup>**

A request is made by means of an application form, which is provided by the Social Insurance Bank.

The second layer is aimed at civil servants, and deals with their side of the procedures, for example:

**General Child Benefit Act, article 17d, sub 3<sup>63</sup>**

The Attorney General will inform the Social Insurance Bank of any circumstances as meant under sub 1 or 2.

In addition to such norms, there are other types of sentences as well. Hart (1961) distinguishes two types of rules in a law: primary and secondary rules. The primary rules are the rules that

<sup>60</sup> Or province, municipality, etc., depending on the jurisdiction of that regulation.

<sup>61</sup> **Algemene kinderbijslagwet, artikel 7, eerste lid**

De verzekerde heeft overeenkomstig de bepalingen van deze wet recht op kinderbijslag voor een eigen kind, een aangehuwd kind en een pleegkind dat

- a. jonger is dan 16 jaar en tot zijn huishouden behoort, of
- b. jonger is dan 18 jaar en door hem in belangrijke mate wordt onderhouden.

<sup>62</sup> **Algemene kinderbijslagwet, artikel 14, tweede lid**

Een aanvraag wordt ingediend door middel van een door de Sociale verzekeringsbank beschikbaar gesteld aanvraagformulier.

<sup>63</sup> **Algemene kinderbijslagwet, artikel 17d, derde lid (vervallen)**

Het openbaar ministerie doet van een omstandigheid als bedoeld in het eerste en het tweede lid mededeling aan de Sociale Verzekeringsbank.

refer to human behaviour. Secondary rules are actually rules about primary rules, and form a meta-level. Three types of secondary rules are given by Hart: rules of recognition, rules of change and rules of adjudication. Rules of recognition determine which rules are ‘official’, rules of change allow for the changing of rules and rules of adjudication empower individuals to judge whether a rule has been broken.

The norms given above are all sentences that represent primary rules, but a law text also contains sentences that represent such secondary rules. For example, this is a sentence containing a rule of change, a sentence that allows others to set new rules:

**General Child Benefit Act, article 24b<sup>64</sup>**

By Ministerial Decree additional rules can be set regarding the articles 24, sub 1, 2, 3, 4, 5 and 6, and 24a.

A law may also contain sentences that describe actual changes to other legislation. These do not correspond to Hart’s rules.

So, now we have a four-layered model of the kind of rules we can encounter in the law, consisting of the core rules and three types of supporting rules.

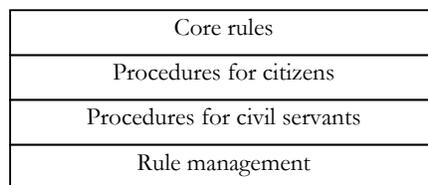


Figure 7: Four-layered model of a legislative text

In addition to these rules, laws contain definitions, sentences that define concepts used elsewhere in the law. These definitions support the rules, both the core rules and the procedures. Together with rules, the definitions make up the body of the law. Add to that the introduction, conclusion and appendices, and we come to a more complete model of a legislative text:

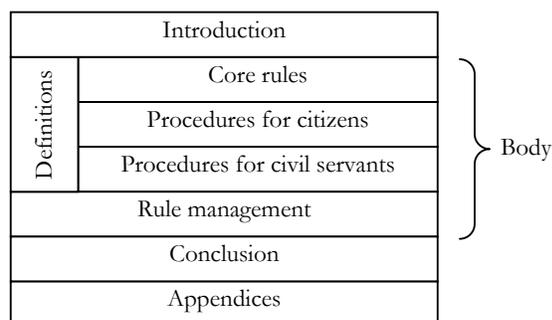


Figure 8: Expanded model of a legislative text

Expert systems that deal with the law are often limited to one or two levels of this model. For example, a system that advises people whether or not they have a right to child benefit will only address the core rules. A somewhat more elaborate system, which also advise them on

<sup>64</sup> **Algemene kinderbijslagwet, artikel 24b**

Bij ministeriële regeling kunnen nadere regels worden gesteld met betrekking tot de artikelen 24, eerste, tweede, derde, vierde, vijfde en zesde lid, en 24a.

what they need to do to obtain them will contain rules from the second layer as well, but is unlikely to include rules from the other layers. On the other hand, a workflow system for the child benefit procedure would include rules from both the second and third layer, and a system that keeps track of the history of a law text would track the rules from the fourth layer. The contents of the introduction and the conclusion are seldom relevant for expert systems, so we not consider those further in this chapter. The contents of the appendices are often relevant, but their contents can vary wildly between laws, so they do not share many common patterns, and we will disregard those as well.

Though the model sketched above does line up neatly with the different expert systems that exist, it does not give us much insight in the types of sentences that we can encounter in a law. A more detailed view is given by Tiscornia and Turchi (1997), who have made a categorisation to use for their *Lexsearch* project. They present two models. The first one is based on the position of the components, and is shown in table 5.

Identifying Elements	Introductory Part	Main Part	Final Part
Type of document	Preamble General	Definitions	Organising financial coordination
Date and number of document	Citations	Attributing competence	Transitional, temporal and/or territorial force
Title	Formula of promulgation Scope General Principles	Constitutive Interpretative Instituting Procedural Sanctioning Derogation Extensions Abrogation Substitution Prorogation Suspension	Date and time of promulgation

Table 5: Tiscornia and Turchi's model of provisions, based on their position

This model is similar to that of de Maat and Winkels above, in that it divides the law into an introduction, body and conclusion. This model disregards the appendices, and separates the identifying elements from the introduction. Also, it is more detailed with regard to the contents of these parts.

The second model re-arranges the provision types according to their function, called “basic components”. It is shown in table 6.

Identifying Elements	Fixed Parts	Provisions	Links
Type of document	Formula of promulgation	Definitions	<i>Amendments to the text:</i> Substitutions, Abrogations and Additions
Date and number of document	Date and place of promulgation	Sanctions	<i>Temporal amendments:</i> Prorogations and Suspensions.
Title		Prescriptions	<i>Substantive amendments:</i> Derogations, Extensions and Interpretations

Table 6: Tiscornia and Turchi's model of provisions, based on their function

Tiscornia and Turchi consider several groups of provisions. The first group is that of the norms that concern the text under examination, which includes *fixed parts*, *scope*, *general principles* and *financial provisions*.

The second group are the norms that describe the relation to the legislative system: *links*, *co-ordinating norms* and *transitional norms*.

The third group is formed by the actual norms: statements that prescribe some behaviour (*prescriptions*) and the statements that describe the *sanctions* that are imposed when the prescriptions are not complied with.

The fourth group is formed by the *constitutive norms*, which Tiscornia and Turchi describe as a non-uniform category, which contains:

- *procedural norms*;
- *classificatory norms*;
- *institutive norms* (norms that create a body or an office that did not exist before);
- *norms attributing competence*;
- *definitions*;
- (true) *constitutive norms*: norms that create a legal effect.

The classification made by Atienza and Manero (1998) features more sub-classifications. In total, they distinguish nineteen different kinds of legal sentences, divided in several groups, as depicted in figure 9.

The first distinction they make is between *legal* and *meta-legal* sentences. Legal sentences are those sentences that belong to some legal system, whereas meta-legal sentences are those sentences that are about such a legal system. These meta-legal sentences correspond to Hart's rule of recognition.

The legal sentences are then divided between sentences of a *practical nature* and of a *non-practical nature*. With practical nature, Atienza and Manero mean that the sentences have the function of guiding or evaluating behaviour. The group of sentences of a non-practical nature consists of definitions, which do not guide behaviour, but that instead identify the meaning of other sentences.

The sentences of a practical nature are divided in *normative* and *evaluative* sentences. Evaluative sentences are sentences that evaluate behaviour, and thus provide a motivation for the *normative* rules that guide behaviour.

The normative sentences fall apart in actual *norms* and sentences that express the use of powers that have been conferred by norms, such as the enacting or repealing of a law. The sentences in this second group are called *normative acts*.

Within the norms, there is a distinction between *regulative* and *constitutive* norms (or *deontic* and *non-deontic* norms). Regulative norms are the norms that actually guide behaviour, whereas constitutive norms state how institutional results and normative changes are brought about. The regulative norms are further divided based on three criteria:

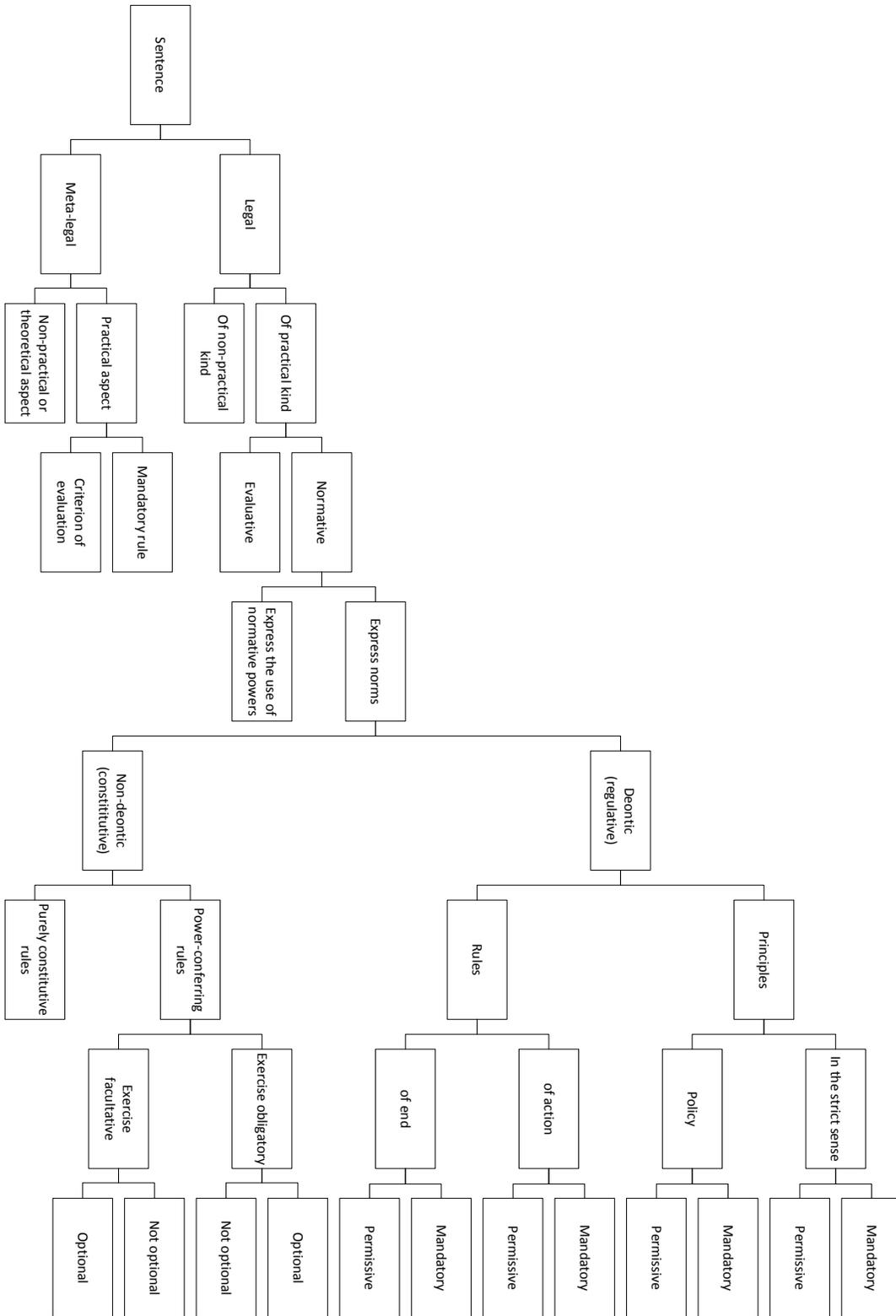


Figure 9: Atienza and Manero's sentence classification

1. whether the conditions for their application are indicated in an open or a closed form (*principles* versus *rules*);
2. whether the norm prescribes an action to be taken or an end state that has to be reached (*strict principles* and *action rules* vs. *policies* and *end rules*);
3. whether the norm indicates obligatory behaviour or facultative behaviour.

Two different groups of constitutive norms are recognised. *Power-conferring rules* are rules that stipulate what one must do to produce an institutional result or normative change. They can make the exercise of those powers obligatory or facultative. Furthermore, Atienza and Manero also distinguish between whether or not executing the action is optional or non-optional<sup>65</sup>.

*Purely constitutive rules* are rules that stipulate that if a certain state of affairs comes to pass, some institutional result or normative change is produced.

The meta-legal sentences are further divided into three groups. First, there is the distinction between sentences of a practical nature and of a non-practical nature, as it has also been applied to legal sentences. The sentences of a practical nature are then divided into *mandatory rules* and *criteria for evaluation*.

The different models suggest that the categories of provisions can be grouped in different ways. This is most clearly illustrated by Tiscornia and Turchi, who group the same provision categories in two different ways. Atienza and Manero consider the motivations for the rules (evaluative sentences) to be closer to the normative sentences than definitions, grouping the first to as sentences of the practical kind, while labelling definitions as sentences of the non-practical kind. In the other models, the definitions are considered to be closer to the normative sentences, as those two together form the actual rules.

The models also suggest that there are many ways to divide the normative sentences. Atienzo and Manero divide the deontic norms along three axis: openness of the condition, describing an action or an end result, mandatory or permissive. De Maat and Winkels distinguish norms by the “distance” to the citizen. Tiscornia and Turchi divide deontic norms in prescriptions and sanctions.

Constitutive norms are divided into purely constitutive norms and power-conferring rules by Atienzo and Manero. Tiscornia and Turchi distinguish six different types: procedural norms, classificatory norms, institutive norms, power-conferring norms, definitions and purely constitutive norms. As mentioned above, Atienzo and Manero do not consider definitions as norms (nor do de Maat and Winkels). The other subdivisions are not named by Atienzo and Manero, and likely fall under their category of purely constitutive norms.

Tiscornia and Turchi also label several amending sentences (which they group together as “links”), which are grouped by Atienzo and Manero under normative acts and by De Maat and Winkels as rule management.

---

<sup>65</sup> Even though it may be obligatory to obtain a certain result, performing the action to obtain that result may be optional, if there are other actions available to achieve that same result.

Our focus is on the classification of sentences that appear in the body of Dutch laws. This means that we can ignore categories that belong to the introduction or conclusion, such as the fixed parts. We also need not look at Atienza and Manero's meta-legal sentences or evaluative legal sentences, as these do not seem to appear in the body of Dutch laws. This leaves us with three broad categories:

1. Normative sentences, which set the rules for people and organisations that live in a country, and which correspond to Hart's primary rules and Atienza and Manero's express norms.
2. Definitions, which clarify the terms used by normative sentences and correspond to Atienza and Manero's sentences of non-practical kind.
3. Lifecycle and maintenance sentences, which are rules about the law, and which enact, modify and repeal existing regulations, corresponding to Atienza's and Manero's purely normative acts.

In the following sections, the different sentences types found for each of these categories are discussed. These have been published before in de Maat and Winkels (2008, 2010). Following the classification, two experiments are presented that deal with the automatic classification of sentences in these categories.

## 4.1 Norms

The actual content of (original) legal sources is formed by norms. Normative sentences may confer rights and permissions or impose duties and obligations. Procedural rules are expressed as norms as well (usually, each step of a procedure is formulated as an obligation).

Research has been performed on how to distinguish these different types of norms (Franssen, 2007)<sup>66</sup>. During this research, it became clear that it was difficult to separate rights and permissions, and that it was likewise difficult to separate duties and obligations. Because of this, the norms have been grouped together in two large groups (obligations and rights) rather than separated further, as this will not benefit automated processing of the law.

### 4.1.1 Obligations

Obligations are the sentences that express a situation that must (or must not) occur, such as:

**Working Conditions Act, article 11<sup>67</sup>**

In his actions on the workplace, the employee is obliged to take care of his own safety and health and that of others, to his ability, and in accordance with his education and the instructions given by the employer.

Though words like *must* and *is obliged* are used in Dutch law, the guidelines recommend not to use these. Instead, expressions are used that describe a desired situation as if is a fact. An example of such norm is:

<sup>66</sup> This research was supervised by the author of this thesis.

<sup>67</sup> **Arbeidsomstandighedenwet, artikel 11**

De werknemer is verplicht om in zijn doen en laten op de arbeidsplaats, overeenkomstig zijn opleiding en de door de werkgever gegeven instructies, naar vermogen zorg te dragen voor zijn eigen veiligheid en gezondheid en die van de andere betrokken personen.

**Funeral Act, article 46, sub 1<sup>68</sup>**

No bodies are buried on a closed cemetery.

The individual steps of a procedure are also expressed in a similar way, for example:

**Voting Act, article J 25, sub 1<sup>69</sup>**

The voter hands the polling card to the chairman of the polling station.

Procedural norms followed the same sentence formats as regular norms, but have a slightly different meaning, as they are part of a larger procedure. For example, the different steps of such a procedure have to be applied in order. This is often not made explicit in the law, and needs to be gathered from the context.

**4.1.2 Rights**

Rights are sentences that describe a situation that may occur, or that confer a specific right to someone:

**Passport Act, article 9<sup>70</sup>**

Within the limits as determined in this law, every Dutchman has a right to a national passport, valid for five years and for all countries.

**4.1.3 Application Provisions**

Application provisions specify situations in which other legislation (usually an article or subsection of an article) does apply. In this way, the application domain of a norm can be extended or restricted (effectively creating an exception to a rule). Eijlander and Voermans (1999) present application provisions as a way to prevent repetitions in the law.

The official guidelines<sup>71</sup> mention three different phrases that can be used:

1. The word *applies* is used if the provision that is being referred to can be applied literally.
2. The phrase *applies correspondingly* is used if the provision that is being referred to cannot be applied literally, but the meaning is still clear.
3. The phrase *applies, with the understanding that ...* is used if the provision that is being referred to cannot be applied literally, and some modifications have to be made for this application.

<sup>68</sup> **Wet op de lijkbezorging, artikel 46, eerste lid**

Op een gesloten begraafplaats worden geen lijken begraven.

<sup>69</sup> **Kieswet, artikel J 25, eerste lid**

De kiezer overhandigt aan de voorzitter van het stembureau de oproepingskaart.

<sup>70</sup> **Paspoortwet, artikel 9**

Iedere Nederlander heeft binnen de grenzen bij deze wet bepaald, recht op een nationaal paspoort, geldig voor vijf jaren en voor alle landen.

<sup>71</sup> **Aanwijzing 83**

1. De uitdrukking "is van toepassing" wordt gebruikt, indien de bepaling waarnaar wordt verwezen, letterlijk kan worden toegepast.
2. De uitdrukking "is van overeenkomstige toepassing" wordt gebruikt, indien de bepaling waarnaar wordt verwezen, niet geheel letterlijk kan worden toegepast, maar misverstand over de toe te passen tekst uitgesloten is.
3. De uitdrukking "is van toepassing, met dien verstande dat ....." wordt gebruikt, indien de bepaling waarnaar wordt verwezen, gedeeltelijk of met wijziging van bepaalde onderdelen moet worden toegepast.

An example of such an application provision is:

**Constitution, article 7, sub 4<sup>72</sup>**

The previous members do not apply to making commercial advertisements.

In addition to these sentences which indicate that some other legislation does apply, there are also statements that indicate that some other legislation do not apply, using the phrase *does not apply*. Often, an application provision that states that another piece of legislation does apply seems to be included to take away any doubts as to whether it ought to apply or not.

**4.1.4 Penalisations**

The violation of some norms will carry punishment in the form of a fine or imprisonment. If this is the case, the law will specify the penalties. In Dutch law, this is usually done through the phrase *is punished with/by*. In the Penal Code, the behaviour that is punished is usually specified in the same sentence:

**Penal Code, article 365<sup>73</sup>**

The civil servant that, by abuse of power, forces someone to do something, not to do something or to allow something, is punished by imprisonment for at most two years or a monetary fine of the fourth category.

In most other laws, the behaviour is specified in some other article that is being referred to:

**Mining Act, article 133, sub 1<sup>74</sup>**

Breaking article 43, sub 2, is punished with a monetary fine of the second category.

The Penal Code is divided into separate sections for crimes and misdemeanours. Other laws will explicitly specify whether the punishable fact is a crime or a misdemeanour by adding a sentence such as:

**Mining Act, article 133, sub 2<sup>75</sup>**

The fact marked as punishable by this article is a misdemeanour.

These sentences always follow this same structure.

**4.1.5 Calculations**

Calculations are sentences determining some value. Often, several of these sentences together form a procedure that should be followed in order to arrive at the correct result. There are many patterns that indicate calculation. For simple assignments, the verb *to amount* is commonly used:

<sup>72</sup> **Grondwet, artikel 7, vierde lid**

De voorgaande leden zijn niet van toepassing op het maken van handelsreclame.

<sup>73</sup> **Wetboek van strafrecht, artikel 365**

De ambtenaar die door misbruik van gezag iemand dwingt iets te doen, niet te doen of te dulden, wordt gestraft met gevangenisstraf van ten hoogste twee jaren of geldboete van de vierde categorie.

<sup>74</sup> **Mijnbouwwet, artikel 133, eerste lid**

Overtreding van artikel 43, tweede lid, wordt gestraft met geldboete van de tweede categorie.

<sup>75</sup> **Mijnbouwwet, artikel 133, tweede lid**

Het in dit artikel strafbaar gestelde feit is een overtreding.

**Income Tax Act 2001, article 8.17, sub 2<sup>76</sup>**

The elderly discount amounts to € 739.

Other than that, almost any verb indicating a mathematical operation can occur, and indicates a calculation, such as:

**Income Tax Act 2001, article 5.87, sub 3<sup>77</sup>**

The travel deduction as established based on the next subparagraphs is reduced by the compensation received for distances travelled with public transport.

**4.1.6 Delegation**

Delegations confer the power to create additional rules to some legal entity. Most often, this power is conferred onto a minister, for the creation of rules that do not require (immediate) involvement of the parliament.

The delegation can allow for the creation of rules:

**Water Pipe Act, article 4, sub 3<sup>78</sup>**

By Order-in-council, rules may be set regarding the obligation of the owner of a water company to do research into the state of the water that is used by him to prepare tap water.

Alternatively, it can be an order to create rules to arrange for something, like:

**Voting Act, article J 34, sub 1, first sentence<sup>79</sup>**

By or based upon Order-in-council, specific rules are set regarding the voting in other ways than by ballots.

Sometimes, a delegation is followed by additional sentences that set additional restrictions or guidelines for the rules that may or must be created. For example, the previous sentence is followed by:

**Voting Act, article J 34, sub 1, second sentence<sup>80</sup>**

As much as possible, these rules are set conform the provisions of this law regarding the voting by ballots.

<sup>76</sup> **Wet inkomstenbelasting 2001, artikel 8.17, tweede lid**

De ouderenkorting bedraagt € 739.

<sup>77</sup> **Wet inkomstenbelasting 2001, artikel 5.87, derde lid**

De op basis van de volgende leden bepaalde reisaftrek wordt verminderd met de voor de per openbaar vervoer afgelegde reisafstand ontvangen reiskostenvergoedingen.

<sup>78</sup> **Waterleidingwet, artikel 4, derde lid**

Bij algemene maatregel van bestuur kunnen regels worden vastgesteld, inhoudende de verplichting van de eigenaar van een waterleidingbedrijf onderzoeken te verrichten met betrekking tot de hoedanigheid van het water dat door hem gebruikt wordt voor de bereiding van leidingwater.

<sup>79</sup> **Kieswet, Artikel J 34, eerste lid, eerste volzin**

Bij of krachtens algemene maatregel van bestuur worden nadere regels gesteld betreffende het stemmen anders dan door middel van stembiljetten.

<sup>80</sup> **Kieswet, artikel J 34, eerste lid, tweede volzin**

Deze regels worden zoveel mogelijk vastgesteld overeenkomstig de bepalingen van deze wet betreffende het stemmen door middel van stembiljetten.

#### 4.1.7 *Publication provision*

A publication provision orders the publication of certain information. It usually accompanies a delegation, ordering the announcement of anything that has been decided based on the delegation.

#### **Animal Feed General Act, article 1, sub 3<sup>81</sup>**

Our Minister announces the enactment of modification of a community measure in the State Gazette, insofar these have to be acted upon, mentioning the articles of this law impacted by the communal regulation.

## 4.2 Definitions

For each term used in a legal text, a clear meaning needs to be established. According to Eijlander and Voermans (2000), there are four techniques to define a term in legal texts:

- a. No definition: the legislator does not define the term at all but uses the common meaning of the word;
- b. Definition by context: a given term may be ambiguous in itself, but put into the context of given legislative text, its meaning is clear and precise;
- c. Definition: a new meaning is given to a term;
- d. Definition by reference: a term is defined by reference, that is, a reference is included to some other text where the term is defined.

This section deals with the definitions: sentences that explicitly define terms that occur in a legal source. Such sentences define terms using other terms, which in turn may be explicitly defined, not defined, defined by context or defined by reference.

### 4.2.1 *Definitions*

Definitions are used to describe the terms that occur in a legal source. A definition mentions both the term being defined as well as the actual definition. In many cases, such a definition is formed by a description of that what is being defined:

#### **General Administrative Law Act, article 1:4, sub 1<sup>82</sup>**

By administrative judge is understood: an impartial body that is appointed by law and charged with administrative judicial settlement.

Another option is a listing of possibilities:

<sup>81</sup> **Kaderwet diervoeder, artikel 1, derde lid**

Onze Minister doet mededeling in de Staatscourant van de vaststelling of wijziging van een communautaire maatregel voorzover daaraan uitvoering moet worden gegeven, onder vermelding van de artikelen van deze wet waarop de communautaire maatregel betrekking heeft.

<sup>82</sup> **Algemene wet bestuursrecht, artikel 1:4, eerste lid**

Onder administratieve rechter wordt verstaan: een onafhankelijk, bij de wet ingesteld orgaan dat met administratieve rechtspraak is belast.

**Telecommunications Act, article 1.1, introduction and item i**<sup>83</sup>

In this law and the stipulations based on it, it is understood by providing of an electronic communications network: building, exploiting, maintaining or making available of an electronic communications network.

Some other forms of definitions can also be identified, such as the abbreviation, which simply abbreviates a longer word or term<sup>84</sup>:

**Tobacco Act, article 1, introduction and item k**<sup>85</sup>

In this law and the stipulations based on it, it is understood by nicotine: nicotine alkaloids.

The most common abbreviation in Dutch law is probably the abbreviation of a specific minister to *Our Minister*:

**Tobacco Act, article 1, introduction and item b**<sup>86</sup>

In this law and the stipulations based on it, it is understood by Our Minister: Our Minister of Health, Welfare and Sport.

Some definitions do not give a description themselves, but instead give a reference to a location where the description is provided (i.e. the term used for definition used is defined by reference). This can either be another source, or a location within the same source:

**Higher Education and Scientific Research Act, article 1.1, introduction and item f**<sup>87</sup>

In this law, it is understood by institutions: an institution as meant in article 1.2.

Next to the term being defined and its definition, many definitions will also contain a scope declaration, which states for which documents the definition is valid. Most often, the scope is *this law* or *this law and the stipulations based on it*. However, more restricting scopes are also possible. For example, the previous example, which defines the meaning of *institutions* within the Higher Education and Scientific Research Act is later overruled for chapter 7 of that act:

**Higher Education and Scientific Research Act, article 7a.1, introduction and item a**<sup>88</sup>

In this chapter, it is understood by institutions: an institution as meant in article 7a.2.

<sup>83</sup> **Telecommunicatiewet, artikel 1.1, aanhef en onderdeel i**

In deze wet en de daarop berustende bepalingen wordt verstaan onder aanbieden van een elektronisch communicatienetwerk: het bouwen, exploiteren, beheren of beschikbaar stellen van een elektronisch communicatienetwerk.

<sup>84</sup> Of course, any definition can be seen as a short-hand notation of a longer description.

<sup>85</sup> **Tabakswet, artikel 1, aanhef en onderdeel k**

In deze wet en de daarop berustende bepalingen wordt verstaan onder nicotine: nicotinealkaloiden.

<sup>86</sup> **Tabakswet, artikel 1, aanhef en onderdeel b**

In deze wet en de daarop berustende bepalingen wordt verstaan onder Onze Minister: Onze Minister van Volksgezondheid, Welzijn en Sport.

<sup>87</sup> **Wet op het hoger onderwijs en wetenschappelijk onderzoek, artikel 1.1, aanhef en onderdeel f**

In deze wet wordt verstaan onder instelling: een instelling als bedoeld in artikel 1.2.

<sup>88</sup> **Wet op het hoger onderwijs en wetenschappelijk onderzoek, artikel 7a.1, aanhef en onderdeel a**

In dit hoofdstuk wordt verstaan onder instelling: een instelling als bedoeld in artikel 7a.2.

### 4.2.2 Type Extensions

Type extensions are very similar to definitions. However, instead of completely defining a new term, they expand or limit an earlier definition. The most common use of type extensions is to expand a *common sense* definition. In these cases, the law source does not define the term, but instead uses the common meaning of the word, and expands upon that meaning. Sometimes, this is done through a “regular” definition, by allowing the term itself to appear in the actual definition:

**Animal Health and Welfare Act, article 1, definition of keeper<sup>89</sup>**

In this law and the stipulations based on it, it is understood by keeper: owner, keeper or herder.

In this case, the second *keeper* does refer to the common sense definition of *keeper*, and not (recursively) to the term *keeper* as defined here. An alternate method for expanding an earlier definition is by using a type extension, which explicitly extends an earlier definition.

**Equal Treatment General Act, article 1, sub 2<sup>90</sup>**

By direct distinction based on gender is also understood distinction based on pregnancy, childbirth and motherhood.

The term that is being expanded can be explicitly defined in an earlier definition, but it is also possible that this is not the case. In this case, the type extension extends the common sense definition of the term.

Next to extending a definition, it is also possible to restrict a definition:

**Automobile and Motorbike Tax Act 1992, article 4, sub 1<sup>91</sup>**

In this law and the stipulations based on it, it is understood by motorbike a motorised vehicle on two wheels, as well as such a motorised vehicle which is attached to a sidecar. By motorbike is not understood a moped as meant in article 1, sub 1, item e, of the Road Traffic Act 1994.

### 4.3 Deeming Provisions

Related to the definitions are the deeming provisions, which introduce a legal fiction. The deeming provision declares one situation to be equal to another situation, in a certain context. If a situation is deemed equal to another situation, then any rules that apply to the latter also apply to the first. For example:

<sup>89</sup> **Gezondheids- en welzijnswet voor dieren, artikel 1, definitie van houder**

In deze wet en de daarop berustende bepalingen wordt verstaan onder houder: eigenaar, houder of hoeder.

<sup>90</sup> **Algemene wet gelijke behandeling, artikel 1, tweede lid**

Onder direct onderscheid op grond van geslacht wordt mede verstaan onderscheid op grond van zwangerschap, bevalling en moederschap.

<sup>91</sup> **Wet op de belasting van personenauto's en motorrijwielen 1992, artikel 4, eerste lid**

In deze wet en in de daarop gebaseerde regelingen wordt verstaan onder motorrijwiel een motorrijtuig op twee wielen, alsmede een dergelijk motorrijtuig dat is verbonden met een zijspanwagen. Onder motorrijwiel wordt niet verstaan een bromfiets in de zin van artikel 1, eerste lid, onderdeel e, van de Wegenverkeerswet 1994.

**Income Tax Act 2001, article 2.2, sub 2, introduction and item a<sup>92</sup>**

A Dutchman who is employed by the State of the Netherlands is always deemed to live in the Netherlands if he is posted as a member of a diplomatic, permanent or consular representation of the Kingdom of the Netherlands in foreign countries.

The effect of this statement is that someone is considered to live in the Netherlands, even though he actually lives outside of the Netherlands.

**4.4 Lifecycle and maintenance****4.4.1 Enactment Date**

These are sentences that set the enactment date for (part of) a regulation, or that arrange for the enactment date to be set.

Each law includes a sentence that set its enactment date, or arranges for its enactment to be set. The most straightforward of these simply set a date for the enactment of the law:

**Fuel Taxes Environment Tariffs Act 1991, article IV, sub 1, first sentence<sup>93</sup>**

This law is enacted starting on January 1st, 1991.

However, this is not very common. The guidelines suggest three common formats, the first of which is to defer the setting of the date to a Royal Decree<sup>94</sup>:

**Exception Situations Coordinating Act, article 11<sup>95</sup>**

This law is enacted on a date to be set by Royal Decree.

The others link the enactment date to the date of publication, either following it directly:

**Act of July 7th, 2010 (Stb. 2010/305), article 9<sup>96</sup>**

This law is enacted starting on the day after the date of publication

Or with some delay:

<sup>92</sup> **Wet inkomstenbelasting 2001, artikel 2.2, tweede lid, aanhef en onderdeel a**

Een Nederlander die in dienstbetrekking staat tot de Staat der Nederlanden, wordt steeds geacht in Nederland te wonen indien hij is uitgezonden als lid van een diplomatieke, permanente of consulaire vertegenwoordiging van het Koninkrijk der Nederlanden in het buitenland.

<sup>93</sup> **Tariefwet brandstofheffingen milieu 1991, artikel IV, eerste lid, eerste zin**

Deze wet treedt in werking met ingang van 1 januari 1991.

<sup>94</sup> In such cases, the Royal Decree will contain a sentence of a similar format as described here to set the enactment date of the law. A Royal Decree which sets an enactment date for a law does not need to set an enactment date for itself; it is automatically enacted at its publication date.

<sup>95</sup> **Coördinatiewet uitzonderingstoestanden, artikel 11**

Deze wet treedt in werking op een bij koninklijk besluit te bepalen tijdstip.

<sup>96</sup> **Wet van 7 juli 2010 (Stb. 2010/305), artikel 9**

Deze wet treedt in werking met ingang van de dag na de datum van uitgifte van het Staatsblad waarin zij wordt geplaatst.

**Wrongful Act Conflict of Laws Act, article 10<sup>97</sup>**

This law is enacted starting on the first day of the second calendar month after the date of publication in the State Gazette in which it is included.

More complicated situations exist in which the date differs for different parts of the law:

**Notaries Act, article 134<sup>98</sup>**

This law is enacted on a date to be set by Royal Decree, which may differ for separate parts and articles.

Or:

**Artificial Insemination Donor Data Act, article 14<sup>99</sup>**

This law is enacted on a date to be set by Royal Decree, which may differ for separate parts and articles, with the exception of article 3, sub 2, second sentence, and sub 3 up to and including 5, which provisions are enacted starting on the date of the first calendar month after two years after the date of publication in the State Gazette in which this law is included.

**4.4.2 Short Title**

If it is thought necessary, a law will also define a short title which can be used to refer to it.

**Notaries Act, article 135<sup>100</sup>**

This act may be referred to as: Notaries act.

It is also possible that a law will modify the short title of another law. This is usually done to avoid confusion, when a new law has the same name as he predecessor.

Sometimes, the short title may be abbreviated, which is indicated by a provision like:

**Income Tax Act 2001, article 11.4, sub 2<sup>101</sup>**

The short title may be abbreviated to: IT Act 2001.

**4.4.3 Change Provisions**

Change provisions are modifications in existing legislation. Most laws are amending laws, consisting mostly of such changes in other laws (instead of new rules). In these laws, change provisions make up the bulk of the text.

<sup>97</sup> **Wet conflictenrecht onrechtmatige daad, artikel 10**

Deze wet treedt in werking met ingang van de eerste dag van de tweede kalendermaand na de datum van uitgifte van het Staatsblad waarin zij wordt geplaatst.

<sup>98</sup> **Wet op het notarisambt, artikel 134**

Deze wet treedt in werking op een bij koninklijk besluit te bepalen tijdstip, dat voor de verschillende onderdelen en artikelen verschillend kan zijn.

<sup>99</sup> **Wet donorgegevens kunstmatige bevruchting, artikel 14**

Deze wet treedt in werking op een bij koninklijk besluit te bepalen tijdstip dat voor de verschillende artikelen of onderdelen daarvan verschillend kan luiden met uitzondering van artikel 3, tweede lid, tweede volzin, en derde tot en met vijfde lid, welke bepalingen in werking treden met ingang van de eerste kalendermaand na verloop van twee jaren na de datum van uitgifte van het Staatsblad waarin deze wet wordt geplaatst.

<sup>100</sup> **Wet op het notarisambt, artikel 135**

Deze wet wordt aangehaald als: Wet op het notarisambt.

<sup>101</sup> Wet Inkomstenbelasting 2001, artikel 11.4, tweede lid

De citeertitel kan worden afgekort tot: Wet IB 2001.

There are four types of changes: insertion of new text, replacing of text, deletion of text and renumbering of sections.

An insertion adds new text to the document. The text describes this as appending text if the text is added to the end of a structure element; otherwise it is described as inserting text.

**Act of June 6<sup>th</sup>, 2002 (Stb. 303), article I, sub IIa<sup>102</sup>**

To article 7.36, a new sentence is appended, to read as follows: Article 7.34, sub 5, applies correspondingly.

When replacing text, some text is removed and new text is added instead. This can be done at the level of a few words within a sentence:

**Act of June 6<sup>th</sup>, 2002 (Stb. 303), article III, sub V<sup>103</sup>**

In article 7.12, sub 1, second sentence, «article 7.3b» is replaced by: article 7.3c.

Alternatively, if an entire sentence, section or article is replaced, the modifying provision will simply refer to that element and quote the new text:

**Act of June 6<sup>th</sup>, 2002 (Stb. 303), article IV, sub B<sup>104</sup>**

Article 2.8 will read: ...

The deletion of text, a repeal, can affect an entire law, an element of a law or only a few words.

**Act of June 6<sup>th</sup>, 2002 (Stb. 303), article I, sub QQ<sup>105</sup>**

Article 17.2 is repealed.

The last change is the renumbering (or relettering) of structure elements. Because renumbering an element requires the modification of all text referring to that element, it is somewhat uncommon for articles to be renumbered. On the other hand, anything below the level of article (subsections and lists) is almost always renumbered to keep a continuous numbering.

**Act of June 6<sup>th</sup>, 2002 (Stb. 303), article IIIc, sub C<sup>106</sup>**

The articles 17a.1 to 17a.25 are renumbered to the articles 17.20 to 17.54.

Often, renumbering is related to the insertion or deletion of text. In the modifying text, the operations are often combined in one sentence:

<sup>102</sup> **Wet van 6 juni 2002 (Stb. 303), artikel I, lid IIa**

Aan artikel 7.36 wordt een volzin toegevoegd, luidende: Artikel 7.34, vijfde lid, is van overeenkomstige toepassing.

<sup>103</sup> **Wet van 6 juni 2002 (Stb. 303), artikel III, lid V**

In artikel 7.12, eerste lid, tweede volzin, wordt «artikel 7.3b» vervangen door: artikel 7.3c.

<sup>104</sup> **Wet van 6 juni 2002 (Stb. 303), artikel IV, lid B**

Artikel 2.8 komt te luiden: ...

<sup>105</sup> **Wet van 6 juni 2002 (Stb. 303), artikel I, lid QQ**

Artikel 17.2 vervalt.

<sup>106</sup> **Wet van 6 juni 2002 (Stb. 303), artikel IIIc, lid C**

De artikelen 17a.1 tot en met 17a.25 worden vernummert tot de artikelen 17.20 tot en met 17.54.

**Registered Partnership Modification Act, article 4, sub A, sub 1<sup>107</sup>**

Renumbering sub 2 to 5 to sub 3 to 6, a new sub 2 is inserted, reading: ...

Another renumbering operation is the adding of an index to a paragraph that did not have one before:

**Act of July 12<sup>th</sup>, 2009 (Stb. 2009/245), article I, sub Da, sub 1<sup>108</sup>**

Before the text, the index «1» is placed.

Renumbering is not always done explicitly; sometimes the header of a structure element is modified in a way that also affects the index, implicitly renumbering the element:

**Act of June 6<sup>th</sup>, 2002 (Stb. 303), article IIIa, sub A<sup>109</sup>**

The heading of chapter 5a will read: Chapter 5. Accreditation in higher education.

None of the sentences above have a complete reference. They refer to articles, but not to articles in a specific law. As described in section 3.3, such an incomplete reference normally refers to another element of the same text, but this is usually not the case in amending laws. If many changes are made in the same text, then they are grouped and preceded by a sentence that sets the scope, such as:

**Act of June 6<sup>th</sup>, 2002 (Stb. 303), article I, introduction<sup>110</sup>**

To the Higher Education and Academic Research Act, the following modifications are made:

## 4.5 Experiment: Pattern-based Approach

We built a classifier (in Java) that takes well-structured legal sources as input and tries to classify their sentences according to their type based on typical patterns associated with these types. For each sentence type, the classifier includes several patterns, as we have found that there were many different ways in which each type of sentence can be expressed. This variation in expressions has also been found by other researchers, such as Deschamps (2011).

Deschamps also discusses semasiological variation. This means that a single expression is sometimes used to for different types of sentences. She gives the example of an obligation, which turns out to be a permission, due to the existence of exceptions to that obligation. We do not take context in consideration when classifying the sentences, so a sentence is classified based on its own pattern and meaning. Thus, the sentence form Deschamps example would be classified as an obligation. Just as with the actual text, the meaning of the entire law only becomes clear when the different sentences are combined.

<sup>107</sup> **Aanpassingswet geregistreerd partnerschap, artikel 4, lid A 1**

Onder vernummering van het tweede tot en met vijfde lid tot derde tot en met zesde lid wordt een nieuw tweede lid toegevoegd, luidende: ...

<sup>108</sup> **Wet van 12 juni 2009 (Stb. 2009/245), article I, lid Da, lid 1**

Voor de tekst wordt de aanduiding «1» geplaatst.

<sup>109</sup> **Wet van 6 juni 2002 (Stb. 303), artikel IIIa, lid A**

Het opschrift van hoofdstuk 5a komt te luiden: Hoofdstuk 5. Accreditatie in het hoger onderwijs.

<sup>110</sup> **Wet van 6 juni 2002 (Stb. 303), artikel I, aanhef**

In de Wet op het hoger onderwijs en wetenschappelijk onderzoek worden de volgende wijzigingen aangebracht:

The classifier assumes that the input is structured using MetaLex XML. In MetaLex, sentences and lists are marked, as well as the separate list items within each list. This enables the classifier to treat each sentence separately.

The classifier is a simple pattern matcher. We used 88 patterns from about twenty Dutch laws. Most patterns consist of only a verb phrase, like *mogen* (may) for a right/permission or *wordt aangehaald als* (is referred to as) for the defining of a short title. Sometimes, additional keywords have been added, as in *kan regels stellen* (may create rules). As additional examples, the patterns for obligations and sentences setting a short title are given in appendix C.

The patterns are stored in a format for the Java pattern matcher (`java.util.regex`). The patterns mentioned above become:

```
\s+(mag|mogen)\b
\s+wordt\s+aangehaald\s+als(?:)\s+
\s+kan\s+regels\s+stellen\s+
```

In this format, `\s+` denotes one or more whitespace characters, `\b` denotes a word boundary, and `(:)?` is an optional colon. The first pattern allows for either the singular or the plural form of the verb.

The classifier will attempt to match a sentence to each available pattern. If the sentence matches several patterns, the classifier will prefer the longest of the matches. (This does not happen often; however, some of the patterns overlap, such as *kan* for a right and *kan regels stellen* for a delegation.)

As mentioned in section 4.1.1, the official guidelines recommend that words like *must* are not used in obligations. Instead, they are formulated using the normative indicative (Šarčević, 1997), a description of the desired situation. For such sentences, no patterns could be identified. However, as this is the only category for which no patterns exist, the parser assumes that whenever a sentence does not contain any pattern, it is a sentence using the normative indicative (and therefore an obligation).

A different approach was tried to classify sentences with an embedded list, such as:

**Tobacco Act, article 1<sup>111</sup>**

In this law, and in the stipulations based on it, is understood by:

- a. tobacco products: ... ;
- b. Our Minister: ...;
- c. appendix: ...;
- ...

Two different approaches were made. The first approach (de Maat & Winkels, 2008) assumed that most lists, like the example above, included the pattern needed for classification in the introduction of the sentence.

In the second approach (de Maat & Winkels, 2009), the classifier searches for a pattern in the introduction first. If one is found, the entire list is classified according to that pattern. If no pattern is found, the individual list items are searched for patterns and classified as if they were separate sentences.

The classifier was tested on eighteen different Dutch regulations, four of which were completely new laws. The others were amending laws, mostly containing changes in existing laws. With the exception of a single Royal Decree, these were all bills, pending at parliament. The length of the laws varied from very short (three sentences) to quite long (166 sentences on 23 pages A4); most were quite recent (patterns in the past have been different).

In each document, all sentences belonging to the body of the text were parsed, including any sentences that were quoted from or to be inserted in other documents. To check whether clauses were classified correctly, all sentences and lists in all laws were also classified manually. This manual classification was performed by several persons, who classified the sentences based on the description given in de Maat and Winkels (2007).

---

<sup>111</sup> **Tabakswet, artikel 1**

In deze wet en de daarop berustende bepalingen wordt verstaan onder:

- a. tabaksproducten: ...;
- b. Onze minister: ...;
- c. bijlage: : ...;
- ...

### 4.5.1 Results

Table 7 shows the results for sentences (not lists) in the different sources.

Source	Sentence			Type
	Total	Correct	%	
Royal Decree Stb. 1945, F 214 (as modified per 01/01/2002)	25	23	97%	New
Bill 20 585 nr. 2	31	30	97%	New
Bill 22 139 nr. 2	22	20	91%	New
Bill 27 570 nr. 4	21	16	76%	Change
Bill 27 611 nr. 2	11	11	100%	Change
Bill 30 411 nr. 2	141	128	91%	New
Bill 30 435 nr. 2	40	39	98%	Change
Bill 30 583 nr. A	27	27	100%	Change
Bill 31 531 nr. 2	3	3	100%	Change
Bill 31 537 nr. 2	29	29	100%	Change
Bill 31 540 nr. 2	7	7	100%	Change
Bill 31 541 nr. 2	8	8	100%	Change
Bill 31 713 nr. 2	7	6	86%	Change
Bill 31 722 nr. 2	31	22	71%	Change
Bill 31 726 nr. 2	78	67	86%	Change
Bill 31 832 nr. 2	7	7	100%	Change
Bill 31 833 nr. 2	4	4	100%	Change
Bill 31 835 nr. 2	99	90	91%	Change
<b>Total</b>	<b>591</b>	<b>537</b>	<b>91%</b>	

Table 7: Sentence classification results per law

The first thing to notice is that the classifier performs well, classifying 90% of all sentences correctly. Out of the 591 sentences, 537 were classified correctly. Table 8 presents the results for the different types of sentences. The column *In corpus* shows the number of sentences present in the test set for each type, both as an absolute number and as a percentage. The column *Missed* shows how many of these sentences were not correctly identified. For example, the test set contained 35 application provisions, but one was incorrectly classified (meaning that 34 were correctly classified). The column *False* presents the amount of sentences that were incorrectly classified as a particular type, e.g. eight sentences were incorrectly classified as an application provision. Each false positive corresponds to a *Missed* somewhere else.

The biggest part of the sentences is formed by the norms. 44% of all sentences belong to one of the norm categories. The next biggest category consists of the change provisions, with 37%. Some sentences were a concatenation of two sentences. For example, one sentence contained two changes: renumbering and a repeal. These sentences are listed in table 8 as ‘Mixed type’.

About half of the misses were caused by patterns that were unknown to the classifier; these sentences were incorrectly classified as the default (normative indicative), and sometimes as a norm of the type right/permission. Two notable patterns were missing: a renumbering pattern dealing with re-lettering rather than renumbering, and a new pattern for delegations.

Type	In corpus	Missed	False	
Definition	2%	12	1	0
Norm - Right/Permission	11%	64	4	13
Norm - Obligation/Duty	5%	29	0	1
Delegation	3%	19	6	0
Publication Provision	1%	4	0	0
Application Provision	7%	40	1	8
Enactment Date	3%	17	1	0
Short Title	1%	4	0	0
Value Assignment	0%	1	0	0
Penalisation	0%	0	0	2
Change	41%	241	16	8
<i>Scope</i>	9%	54	0	0
<i>Insertion</i>	7%	44	1	0
<i>Replacement</i>	19%	111	4	0
<i>Repeal</i>	4%	23	7	8
<i>Renumbering</i>	2%	9	4	0
Mixed Type	1%	3	3	0
Norm - normative indicative (default)	27%	157	23	23
<b>Total</b>		591	55	55

Table 8: Sentence classification results per sentence type

Those misclassifications that were not caused by missing patterns were instead caused by patterns that were somehow too broad. For example, most false positives of the “repealed” type sentences were provisions concerning the repeal of fines instead of articles. This will require more sophisticated patterns or dedicated ‘anti-patterns’ (i.e. not applicable when it contains the word ‘fine’).

Both false penalisations were in fact a right; the pattern that triggered this classification was part of a qualification of a legal body that was given certain rights. Such a qualification is given in an auxiliary sentence. This means that classifier will find two (or even more) patterns: one in the auxiliary sentence, and one in the principal sentence. As it does not have the option to distinguish between the two, it will pick the longest match (which will not always be the correct one).

If the principal sentence does not contain any pattern (because it uses the normative indicative), the classifier will only find the pattern in the auxiliary sentence, and will automatically arrive at the wrong conclusion. This is the cause of almost all false rights and false application statements.

Table 9 shows the distribution of the patterns that were actually encountered in the test set. For each type, the number of patterns known is shown, as well as the number of patterns encountered. The results column shows for each encountered pattern how often it has been correctly applied, and how often it has been incorrectly applied, causing a false positive result.

Type	Patterns Known	Patterns Used	Results per pattern	
			Correct	False
Definition	14	5	6	0
			2	0
			1	0
			1	0
			1	0
Norm – Right/Permission	17	3	55	13
			4	0
Norm – Obligation/Duty <sup>112</sup>	15	8	1	0
			6	1
			5	0
			3	0
			2	0
			2	0
			1	0
			1	0
Delegation	7	5	5	0
			4	0
			2	0
			1	0
			1	0
Publication Provision	1	1	4	0
Application Provision	5	5	36	5
			2	0
			1	0
			0	1
			0	2
Enactment Date	1	1	16	0
Short Title	2	2	3	0
			1	0
Value Assignment	8	1	1	0
Penalisation	3	1	0	2
Change - Scope	2	2	49	0
			5	0
Change - Insertion	4	4	22	0
			18	0
			2	0
			1	0
Change - Replacement	3	3	66	0
			40	0
			1	0
Change - Repeal	2	1	16	8
Change - Renumbering	3	2	4	0
			1	0
		44	394	32

Table 9: Patterns used

<sup>112</sup> Eight obligations that do not follow the “normative indicative” format were not classified using a pattern (but they were still correctly classified as an obligation). Hence, this table shows only 21 correctly applied patterns for obligations, even though 29 obligations were correctly classified according to table 8.

The numbers suggest that there are a couple of main patterns that account for a majority of the sentences identified. For example, 60 rights were correctly identified. Of those, 55 used the pattern *may* and four used the pattern *is qualified*. This corresponds to the result of Franssen (2007), who concluded that the majority of right could be identified with those two patterns. This distribution suggests that some of the other patterns may be superfluous.

Table 9 also shows that the most false positives are caused by a small set of patterns as well, with one pattern for rights, one for repeal and one pattern for application provisions being the biggest offenders. However, these patterns are also responsible for many of the correct classifications, so removing them will not improve the results. It may be possible to narrow them down instead, but this also carries the risk of reducing the number of correct results.

As said, the results discussed above only referred to the sentences encountered in the regulations. For lists, the performance is influenced by the manner in which they are handled. Table 10 shows the results for the two methods that were tried: classification based on the introduction, and classification based on the introduction followed by items if the introduction contained no pattern.

	<i>Introduction only</i>			<i>Introduction, then Items</i>				
	Total	Correct	%	Total	Correct	Partial	%	
Royal Decree Stb. 1945, F 214 (as modified per 01/01/2002)	4	3	75%	4	4	0	75%	New
Bill 20 585 nr. 2	4	3	75%	4	3	1	75%	New
Bill 22 139 nr. 2	2	2	100%	2	2		100%	New
Bill 27 611 nr. 2	1	0	0%	1	1		100%	Change
Bill 30 411 nr. 2	25	24	96%	25	20	3	80%	New
Bill 30 435 nr. 2	4	4	100%	4	3	1	75%	Change
Bill 31 537 nr. 2	2	2	100%	2	2	0	100%	Change
Bill 31 713 nr. 2				2	2	0	100%	Change
Bill 31 722 nr. 2				6	5	0	83%	Change
Bill 31 726 nr. 2				2	1	1	50%	Change
Bill 31 832 nr. 2				3	3		100%	Change
Bill 31 835 nr. 2				7	4	3	57%	Change
<b>Total</b>	<b>42</b>	<b>38</b>	<b>90%</b>	<b>62</b>	<b>50</b>	<b>9</b>	<b>81%</b>	

Table 10: List classification results per law

Classification on introduction only gives an accuracy of 90%, which is close to the result achieved for sentences. Classification of introduction followed by items performs a lot worse, with an accuracy of only 81%. However, this disregards the lists that were partially correct, i.e. some of the items were classified correctly, while others were not. Using classification by introduction, followed by items, only 5% of the lists were fully classified incorrectly.

#### 4.6 Experiment: Machine Learning Approach

As an alternative to the pattern-based approach for classification, we have attempted a machine learning approach, as literature on general text classification (such as Sebastiani, 2002)

suggests that machine learning is superior to knowledge based approaches. The results presented in this section have been published before in de Maat, Krabben and Winkels (2010).

Machine learning techniques build a classifier by learning the characteristics of each category from a set of already classified examples. In general, these techniques are more flexible and less domain dependent. They also require less expert knowledge, as experts often find it easier to classify something than to come up with reasons for the classification. A disadvantage of machine learning is that it works like a black box; the classifier cannot provide reasons for the classifications it makes. Furthermore, machine learning requires a large dataset to be available, while a knowledge-based approach does not need any formal dataset.

In the legal domain, Francesconi and Passerini (2007) used ML to perform a classification task on provisions in Italian laws. They arrive at similar results as we did (93% accuracy).

Gonçalves and Quaresma (2005) used machine learning for classifying documents from the Portuguese Supreme Court and Attorney General's Office and also achieve accuracy rates above 90%. Opsomer et al. (2009), however, report results no higher than 65% for classifying Belgian environmental laws. These studies are difficult to compare to our own research, as the jurisdictions, fields of law and languages differ. Also, the grain size of the "documents" to be classified ranges from sentences (in our case) to entire chapters of laws (in Opsomer's case). Gonçalves and Quaresma classify court decisions instead of laws.

For the classification of sentences, we use Support Vector Machines (Cortes & Vapnik, 1995). SVMs have been shown to perform at least as good and usually much better in text categorization than other popular algorithms such as decision trees or naive Bayes classifiers (see Yang & Liu, 1999 and Sebastiani, 2002). SVMs were also used by Francesconi and Passerini, Opsomer et al., and Gonçalves and Quaresma mentioned above.

We have done two sets of experiments, using the `libsvm` toolkit for WEKA (Hall et al., 2009), with linear kernel and default settings. The first experiments were aimed at finding the optimal settings for data representation and pre-processing. After those optimal settings had been selected, we evaluated whether a classifier trained using these settings could generalise to texts that were not included in the training set.

We will first discuss the data representation, followed by a description of the two experiments.

#### **4.6.1 Data representation**

Each sentence needs to be represented in a way that can be handled by the automated classifier. A common approach is to represent a document (in our case: a sentence) by the words it contains, disregarding the order in which they appear. This is called a bag of words model. The basic approach is to select all the words as they appear in the sentence, though some changes can be made to the selection process that may lead to better results.

A *stop list* contains those words that are deemed unlikely to be useful in the classification process. It contains words like *the*, *from* and *him*. When using the stop list, words appearing on the list are not included in the representations of the sentences.

*Stemming* is the conversion of a word to its morphological root<sup>113</sup>. When using stemming, the morphological root of words is added to the representations instead of the words themselves.

*Grouping of numbers* is the replacing of numbers by a special character. So, rather than indicating that a sentence includes the number 12 and that another sentence includes the number 182, it is merely noted that both include a number. This may help classification if it is the presence of a number that is an indication of a certain type, rather than its exact value.

*Conversion to lower case characters* means that all upper case characters in a word are replaced by their lower case equivalent.

A *minimal term frequency* adds a threshold for word frequency. When applying a minimal term frequency, words are only included in the representations if they apply at least a certain number of times in the training sentences of one class.

The selected words form a vocabulary  $v$ . This vocabulary contains all the words that appear in the dataset. Each sentence is represented as a vector  $w_1, \dots, w_n$  (where  $n$  is the number of words in the database). Each weight corresponds to a word from the vocabulary. The value of the weight depends on the weighing method used. We tested three commonly used methods:

- A *binary* weight that indicates whether or not a word appears in the sentence. In this case, the weight is one if the word appears in the sentence, and zero if not.
- A *term frequency* (TF) weight that is equal to the number of times that a word appears in a sentence.
- An *inverse document frequency weight* (TFIDF), which incorporates information on how often a word appears in the entire corpus. The weight is equal to:

$$\text{term frequency} \times \log \frac{\text{number of sentences}}{\text{number of sentences containing } w}$$

This weight takes into consideration that words appearing in many sentences are less discriminating.

#### **4.6.2 First Experiment: Pre-processing**

In this experiment, different configurations were tested in order to compare the effect of the different settings. We used the same dataset as the one used in testing our pattern-based classifier<sup>114</sup> (see section 4.5), using only the sentences. Moreover, sentences of classes that were too small were left out, as were sentences with mixed types, i.e. two or more classifications because of auxiliary sentences. This left 584 sentences for the experiment.

Because of the relatively small data set, we used cross-validation for evaluating our classifiers, i.e. use the same data for training and testing. We used a special form of cross-validation: the Leave-One-Out (LOO) procedure. In this procedure, each instance is selected for testing once and evaluated on a classifier based from the training set of all other instances. The results of these tests are shown in table 11. The LOO Accuracy is calculated using the results of all tests combined, and is a predictor for the accuracy of a classifier based on the entire set.

<sup>113</sup> We used the Dutch version of the Snowball stemmer (<http://snowball.tartarus.org/>).

<sup>114</sup> In this section, we will refer to the pattern-based classifier as the knowledge engineering or KE classifier.

	Term weight	Stop list	Group numbers	Stemming	Min. term frequency	Lower case	LOO accuracy
baseline	binary	no	no	no	1	no	93.32
baseline + TF	TF	yes	no	no	1	no	92.29
baseline + TFIDF	TFIDF	yes	no	no	1	no	93.32
baseline + stop list	binary	yes	no	no	1	no	94.01
baseline + grouping	binary	no	yes	no	1	no	92.81
baseline + stemming	binary	no	no	yes	1	no	92.47
baseline + min. term frequency 2	binary	no	no	no	2	no	93.15
baseline + min. term frequency 3	binary	no	no	no	3	no	92.47
baseline + lowercase	binary	no	no	no	1	yes	93.15
<i>Optimal</i>	<i>binary</i>	<i>yes</i>	<i>no</i>	<i>no</i>	<i>2</i>	<i>no</i>	<i>94.69</i>
baseline + stop list + min. term frequency 2							

Table 11: LOO Accuracy for different data representation settings

Even without any pre-processing, the (predicted) accuracy is already quite high, above 93%. The different forms of pre-processing only improve this by small amounts, and often even decrease the performance (when used on their own). Nevertheless, using a stop list has a positive effect on the LOO accuracy.

The highest LOO accuracy as achieved by using binary weight, a stop list and a minimum term frequency of ‘2’, which resulted in a LOO accuracy of 94.69%<sup>115</sup>. Table 12 shows the results for the different classes when using these settings. It shows that the classifier has trouble with definitions, misclassifying 6 out of 14. Also, more than half of the misclassified sentences is classified as an obligation (17 out of 31).

	In corpus	Missed	False	Recall	Precision
Definition	14 2%	6	4	57.14%	66.67%
Permission	59 10%	5	7	91.53%	88.52%
Obligation	181 31%	9	17	95.01%	91.01%
Delegation	19 3%	2	0	89.47%	100%
Publication provision	6 1%	1	0	83.33%	100%
Application provision	41 7%	4	2	90.24%	94.87%
Enactment date	18 3%	1	1	94.44%	94.44%
Short title	4 1%	1	0	75.00%	100%
Change – Scope	55 9%	0	0	100%	100%
Change – Insertion	44 7%	0	0	100%	100%
Change – Replacement	111 19%	0	0	100%	100%
Change – Repeal	23 4%	0	0	100%	100%
Change – Renumbering	9 2%	2	0	77.78%	100%
<b>Total</b>	<b>584</b>	<b>31</b>	<b>31</b>	<b>94.69%</b>	<b>94.69%</b>

Table 12: Results per class when using optimal configuration

A more detailed analysis is made using the confusion matrix, shown in table 13. There we can see that the problem with definitions resides in an inability to distinguish between definitions

<sup>115</sup> Francesconi and Passerini (2007) report similar best settings: replacing digits and non-alphanumeric characters, stemming, binary weighting and a minimum term frequency of 2, leading to a LOO accuracy of 92.44%.

and obligations: all six misclassified definitions are classified as an obligation, while three of nine misclassified obligations are classified as definitions. A similar problem exists between obligations and permissions, with four out of five misclassified permissions classified as obligations, and four out of nine misclassified obligations classified as permissions.

	Definition	Permission	Obligation	Delegation	Publication provision	Application provision	Enactment date	Short title	Change – Scope	Change – Insertion	Change – Replacement	Change – Repeal	Change – Renumbering
Definition	<b>8</b>		6										
Permission		<b>54</b>	4			1							
Obligation	3	4	<b>172</b>			1	1						
Delegation		1	1	<b>17</b>									
Publication provision			1		<b>5</b>								
Application provision	1		3			<b>37</b>							
Enactment date		1					<b>17</b>						
Short title		1						<b>3</b>					
Change – Scope									<b>55</b>				
Change – Insertion										<b>44</b>			
Change – Replacement											<b>111</b>		
Change – Repeal												<b>23</b>	
Change – Renumbering			2										<b>7</b>

Table 13: Confusion matrix when using optimal configuration

#### 4.6.3 Second Experiment: Generalisation Across Laws

Different laws often have a (slightly) different way of phrasing certain sentences. This means that, although within a certain law all definitions follow the same patterns, another law may use different patterns. Often, these patterns are slight variations of each other. This means that a classifier, when faced with a new law, may not perform as well.

Our first experiment does not give a good indication for the generalisation across laws. In the LOO setup, each sentence was tested using a classifier based on all other sentences in the corpus. This means that the training set also contained sentences from the same law as the sentence being tested. In order to test the classifier’s generalisation across laws, another experiment was conducted, this time testing all the sentences in one law using a classifier based on sentences from other laws in the corpus.

For each law, table 14 shows the number of sentences (size), the number of misclassifications in the original LOO experiment (Original LOO) and the number of misclassifications when testing the law using a classifier based on the other laws (Train/Test). Since the dataset for the Train/Test condition is smaller, the results are expected to be a bit worse. However, if the results are a lot worse, this may be because the law used phrasings that did not occur in the training set. Bill 30 411 is an example of a document that does a lot worse with a classifier trained on all other laws (28 misclassified versus 7 in original setup).

To confirm this, we conducted a separate LOO experiment for each law, using only the sentences of that law as dataset (shown in the last column of table 14). For two bills (30 411

and 31 722) this classifier is predicted to perform better than the classifier based on all other laws, suggesting that these two laws do indeed have some phrases that are unique to them. These seem different enough that a classifier trained on other laws will not automatically generalise to include them.

Law	Size	Nr. misclassified sentences		
		Original LOO	Train/Test	One law LOO
Royal Decree Stb.1945, F 214	28	6	8	11
Bill 20 585 nr. 2	31	4	7	13
Bill 22 139 nr. 2	22	1	2	10
Bill 27 570 nr. 4	21	2	5	7
Bill 27 611 nr. 2	11	0	0	5
Bill 30 411 nr. 2	141	7	28	22
Bill 30 435 nr. 2	40	3	3	10
Bill 30 583 nr. A	26	0	1	5
Bill 31 531 nr. 2	3	1	1	3
Bill 31 537 nr. 2	23	0	0	4
Bill 31 540 nr. 2	7	1	1	5
Bill 31 541 nr. 2	8	0	0	3
Bill 31 713 nr. 2	7	0	0	7
Bill 31 722 nr. 2	32	1	9	7
Bill 31 726 nr. 2	78	3	6	7
Bill 31 832 nr. 2	7	1	1	2
Bill 31 833 nr. 2	4	1	1	4
Bill 31 835 nr. 2	99	0	7	7

Table 14: Number of misclassified sentences when testing each law in the dataset separately

In a third experiment, we built a classifier using the entire data set of the previous experiments, and two new laws were used to test the classifier. Both are recent bills, pending at Parliament. One bill (32 393, from May 2010) is an amending law, containing only changes in other laws. The second bill (32 398, from June 2010) is a new law, though it also describes many changes in existing laws.

In this experiment, we also included sentences of mixed type (which cannot be classified correctly) and lists, which were classified based on their headers.

Both new laws were classified by the ML classifier as well as the KE classifier. The results are shown in table 15. Both classifiers performed well, with the ML classifier scoring slightly better on Bill 32 393, but quite a bit worse on Bill 32 398.

Test set	#	ML approach		KE approach		
		Nr. misclassified	Accuracy	Nr. misclassified	Accuracy	
Bill 32 393 nr. 2	sentences	71	3	95.77%	4	94.37%
	Lists	18	3	83.33%	1	94.44%
Bill 32 398 nr. 2	sentences	205	23	88.78%	9	95.61%
	Lists	9	0	100%	0	100%

Table 15: Accuracy in the second experiment

Table 16 shows the confusion matrix for Bill 32 398. The results are quite similar to those of the first experiment, showing problems with the classification of definitions and with the distinction between permissions and obligations.

	Definition	Permission	Obligation	Delegation	Application provision	Enactment date	Short title	Change – Scope	Change – Insertion	Change – Replacement	Change – Repeal	Change – Renumbering
Definition	<b>1</b>		2									
Permission		<b>19</b>	11	1								
Obligation		3	<b>47</b>									
Delegation		3		<b>2</b>								
Application provision			2		<b>7</b>							
Enactment date						<b>1</b>						
Short title							<b>1</b>					
Change – Scope								<b>16</b>				
Change – Insertion									<b>5</b>			
Change – Replacement										<b>71</b>		
Change – Repeal											<b>11</b>	
Change – Renumbering												<b>1</b>
Mixed type											1	

Table 16: Confusion matrix for Bill 32 398 nr. 2

#### 4.6.4 Analysis

As a trained SVM is in essence a black box, it is not possible to determine with absolute certainty why certain misclassifications occur. However, there are some shared features between the misclassifications that make for possible and plausible explanations of the errors. First of all, the ML classifier seems to suffer from two problems that also led to problems for the KE classifier:

- *Keywords appearing in subordinate sentences:* sometimes, keywords that are strongly linked to one of the classes appear in an auxiliary sentence (where it bears no influence on the type of the main sentence). For example, the sentence:

*In that case, he notifies Our Ministers of his plans in advance, so that he can bear their opinion<sup>116</sup>.*

This sentence forms an obligation, but the word *can* in the auxiliary sentence is strongly related to a permission. For the KE approach, we already suggested that filtering out subordinate sentences could help improve the results; it seems this may also help the ML approach.

- *Missing standard phrases:* As was noted in earlier research on the KE classifier, the accuracy of the classifier when dealing with a new law can drop quickly if that law uses a phrase that was not encountered before.

A third problem that occurred with the KE classifier was caused by variations on known patterns that used the same words but a different word order, or that insert other words between the words that make up the actual pattern. It is generally acknowledged that a ML approach is more flexible and is therefore better when dealing with such variations. In our test, the ML classifier did not have problems with these variations.

<sup>116</sup> Hij stelt in dat geval Onze Ministers vooraf in kennis van zijn plannen, ten einde hun mening daaromtrent te kunnen vernemen.

- *Keywords linked to different classes:* Certain keywords are linked to different classes. For example, *may* is an indicator of a permission, but *may not* indicates an obligation. The KE classifier recognises both patterns, and chooses between them based on which pattern is the longest. The ML seems to make more errors in this situation. This may improve with more training data.
- *Normative indicative:* In Dutch law, many obligations are written in the normative indicative, a sentence without any clear keywords. The KE classifier deals with statements of fact by using them as a “default” outcome, assigned to any sentences that do not contain a pattern that signifies another class. It would seem that the ML classifier links the signal words for a passive sentence (the Dutch verb *worden*) to obligations, causing non-obligation sentences that use that verb to be classified as obligations.
- *Not enough data:* In some cases, a standard phrase was present in our training set, but was filtered out due to the minimal term frequency of two. For example, in our first experiment, the only two misclassifications within all ‘changes’ were two renumberings. These sentences were actually reletterings and therefore the only two sentences that contained the Dutch word *verletterd*. Since a minimal term frequency of 2 was used, whenever one of the two reletterings was used as test item the word *verletterd* appeared only once in the training set and therefore was not extracted, and hence not used to classify the sentence.

It also seems that classes with fewer documents, and therefore poorer statistics, lead to more errors (but not always). Francesoni and Passerini report similar results.

- *Sparse data:* Typically, statistical text classification methods not only work better when trained on *more* data, but also on larger documents (although Opsomer et al. blame the large and therefore heterogeneous documents in his case for the poor classification performance). Sentences are small units to be classified and therefore the document vectors are almost empty.
- *Focus on the wrong keywords:* The KE classifier will only focus on keywords within the patterns provided. A ML classifier ‘picks’ its own keywords. This is usually an advantage, since it can discover new correlations. However, in our experiments, the classifier sometimes discovered correlations within the test set that do not generalise to new data. For example, one training set might by coincidence contain a lot of permissions for some advisory board. In this case, the ML classifier could link the words *advisory* and *board* to a permission, causing future non-permissions containing these words to be wrongfully classified as a permission. It is possible that this problem is caused by *overfitting*. SVMs are supposed to be resistant to overfitting (Joachims, 1998), but they are not immune.

In all the above experiments, no limit was set to the number of extracted terms from the training set; between 700 and 1700 terms were extracted in the above experiments. To investigate the question whether overfitting took place, some of the experiments were rerun with a maximum number of selected terms set to 50 words per class, leading to extraction of 250 up to 500 words. The results show a decrease in train accuracy and increase in test accuracy from the original experiments, suggesting that overfitting indeed took place when no limit was set to the amount of selected terms.

Franseconi and Passerini and Gonçalves and Quaresma used the same data set for training and testing (through cross-validation or LOO testing). None of these studies investigated the classification on completely new test sets that were not involved in the building process of the classifier. Franseconi and Passerini reported LOO testing with usually perfect accuracy on the training set, which could indicate that the classifier is overfit to the data. Gonçalves and Quaresma (2005) reported a reduced cross-validation accuracy due to the reduction of selected features from nearly perfect (up to 99.5%) to levels between 93 and 96%. Although this is taken to be a loss in general accuracy, it can also be an indication that the high accuracy levels were a result of overfitting to characteristics specific to the data set that was used. Further research is required to test this hypothesis.

- *Keywords outside of a standard phrase*: Many sentences can be classified based on some standard phrase. Since the ML classifier does not consider the order in which words appear, it is possible that it classifies based on one word of the phrase, rather than the complete phrase. Here, the rigidity of the KE classifier seems to be an advantage rather than a disadvantage.
- *Skeeness*: Machine learners tend to favour bigger classes and prevalent patterns over the more uncommon patterns or smaller classes, which has a general positive effect if the distribution of classes and used patterns in the train data represents the real-world distribution. A negative effect is that uncommon patterns or classes are sometimes misclassified because of their small prior chance. This may be an explanation for the many sentences that are misclassified as an obligation.

The problems mentioned above can be used to explain most errors found, but there remain some errors for which we do not have an explanation yet.

## 4.7 Conclusion

The sentences in Dutch laws can be divided in different categories that each perform a distinct type of knowledge or operation. This classification can be applied automatically, using a classifier that is based on either a pattern-based approach or a machine-learning approach. In our experiments, both achieve an accuracy of over 90%, and it seems possible to improve this, by filtering out auxiliary sentences and by using a larger training set.

At first glance, the SVM classifier seems to perform better than the pattern-based approach, achieving a LOO accuracy of 94.7% on the test set, while the pattern-based classifier scores only 90.7%. However, the result of the pattern-based classifier was heavily impacted by some missing patterns: patterns that were used by a law in the test set that were not used by any laws in the training set. In the LOO setup, this was no (great) issue for the SVM classifier, as the training set would always contain sentences from the same law as the sentence being tested (and was therefore more likely to include the necessary pattern).

When it comes to generalisation to laws outside the training set, the SVM classifier seems to score worse than the pattern-based approach. However, this may improve when a bigger training set is reached.

At the moment, it seems that both methods can achieve rather similar results. We expect that, if we filter the subordinate sentences out of the input, the main difference will be in the classification of those sentences that are misclassified: the KE approach tends to misclassify as an obligation (using the default in case of a missing pattern), whereas the ML approach has more diverse misclassifications (though still focusing on obligations, likely due to skewness).

Our classification contains fifteen different classes. Compared to classifications such as that of Sartor (2007), Tiscornia and Turchi (1997) and Atienza and Manero (1998), the classification distinguishes few subclasses of the actual norms, though a comparison with Francesconi and Passerini (2007) suggests that this may be as detailed a classification as can be made using only surface features of the sentence.

Francesconi and Passerini classify provisions (which may be composed of several sentences) in ten categories, which fall in the same broad categories as we have found, though there are differences in different sub-types distinguished.

In our classification, we only detect two types of basic norms: permission and obligation. Francesconi and Passerini are more detailed and detect permission, duty, prohibition and exception. Due to the normative indicative construction, duty and prohibition are difficult to distinguish in Dutch law. With regard to exceptions, we have chosen to focus on the deontic nature of the exception (i.e. permission or obligation) rather than its status as an (explicit) exception.

As for the rule management class of norms, Francesconi and Passerini distinguish four modifications: repeal, insertion and substitution, which means that they do not have the renumbering operation or the scope declaration. The scope declaration is not an actual operation, but merely a sentence that reduces repetition. It does not form a provision all by itself, so it makes sense that it does not appear in Francesconi and Passerini's classification of provisions. As for renumbering, this operation apparently does not exist in Italian laws. Likewise, there are no provisions to set a short title or the enactment date. On the other hand, Italian law uses delegification provisions, which are part of a process to simplify Italian law, and which make a law susceptible to change by lower sources. This is an operation that does not occur in Dutch law.

Another example of the differences between jurisdictions can be taken from Ogawa, Inagaki and Toyama (2008), who classify modifying sentences for an automated consolidation system. They recognise three types of renumbering operations: renumbering, attaching and shifting. Renumbering and attaching (adding an index to previously unnumbered text) correspond to our renumbering category. Shifting, however, does not occur in Dutch laws. It is a renumbering operation that does not explicitly specifies the target numbering, but instead specifies how many positions the articles are “shifted”, e.g. *Articles from 22 to 27 shall be shifted down two articles at a time.*

When attempting to classify the actual norms in the law, we initially focussed on distinguishing the different deontic norms and power-conferring norms. We found only patterns to distinguish between obligations and rights; no patterns were found that indicated power-conferring norms. We did find patterns for one subcategory of power-conferring norms,

namely delegation of legislative powers, and the related publishing provision. It may be possible that more patterns can be identified by focussing on such sub-category rather than the entire category. This may be one method to improve the current classifier.

Moreover, the ML classifier has been trained using a dataset classified using the classes for which patterns had been found. It may be that a ML approach can detect more classes than the KE approach, so it may be interesting to train a ML classifier based on a dataset that is classified using a more detailed classification.

A way to increase the performance of the classifiers may be to combine the two approaches. For example, rather than choosing one approach over the other, both classifiers can be run, with an additional module that decides which classification to use in case the outcome of the classifiers does not match. Another way to combine the approaches is by using the presence of the patterns in a sentence as additional features for the SVM, which may improve its performance.

If we consider the two methods as being more or less equal in terms of accuracy, then the pattern based approach is preferable, as it gives us more information on the reason for a given classification. This information is useful if when we proceed to make a model of the sentence (see the next chapter). However, if in the future, we know more exactly what information we need for this modelling, it may be possible to create a SVM that gives this information in addition to the classification, creating a level playing field.